**Problem1:**

For the Greeks Calculation part, we're going to use two methods: GBSM and finite difference derivative based on the in-class notes:

| Greek | Definition | Formula GBSM |
|---|---|---|
| Delta, Δ | First Derivative of Price with respect to underlying price, $\frac{\delta P}{\delta S}$ | • Call: $e^{(b-r)T}\Phi(d_1)$ <br> • Put: $e^{(b-r)T}(\Phi(d_1) - 1)$ |
| | | |
| Gamma, Γ | $\frac{\delta^2 P}{\delta S^2}$ Also called Convexity | $\frac{f(d_1)e^{(b-r)T}}{S\sigma\sqrt{T}}$, $f(x)$ is the normal PDF |
| Vega | $\frac{\delta P}{\delta \sigma}$ | $Se^{(b-r)T}f(d_1)\sqrt{T}$ |
| Theta, Θ | $-\frac{\delta P}{\delta T}$ Derivative is negative but often expressed as a positive number. Also called Theta Decay | • Call: $-\frac{Se^{(b-r)T}f((d_1)\sigma}{2\sqrt{T}} - (b-r)Se^{(b-r)T}\Phi(d_1) - rXe^{-rT}\Phi(d_2)$ <br> • Put: $-\frac{Se^{(b-r)T}f((d_1)\sigma}{2\sqrt{T}} + (b-r)Se^{(b-r)T}\Phi(-d_1) + rXe^{-rT}\Phi(-d_2)$ |
| Rho, ρ | $\frac{\delta P}{\delta r}$ Note, formulas are for Black Scholes where rf=b | • Call: $TXe^{-rT}\Phi(d_2)$ <br> • Put: $-TXe^{-rT}\Phi(-d_2)$ |
| Carry Rho | $\frac{\delta P}{\delta b}$ | • Call: $TSe^{(b-r)T}\Phi(d_1)$ <br> • Put: $-TSe^{(b-r)T}\Phi(-d_1)$ |

After converting the formulas into python code, we can get the outputs as shown below:

GBSM Greeks:

| Greek | Call | Put |
|---|---|---|
| Delta | 0.072398 | -0.924346 |
| Gamma | 0.015394 | 0.015394 |
| Vega | 6.156924 | 6.156924 |
| Theta | -7.068720 | -5.682323 |
| Rho | 0.933921 | -13.478033 |
| Carry Rho | 0.958620 | -12.239253 |

Finite Difference Greeks:

| Greek | Call | Put |
|---|---|---|
| Delta | 0.072398 | -0.924346 |
| Gamma | 0.015394 | 0.015390 |
| Vega | 6.156924 | 6.156924 |
| Theta | -7.066234 | -5.679833 |
| Rho | -0.024699 | -1.238780 |
| Carry Rho | 0.958620 | -12.239253 |

After that, we're going to implement the binomial tree to get the option values with and without dividends and the results are shown like below:

American Option Values:

| Type | Call Value | Put Value |
|---|---|---|
| Without Dividends | 0.581716 | 20.791972 |
| With Dividends | 0.461922 | 21.628929 |

Call sensitivity = 0.581716 – 0.461922 = 0.119794

Put sensitivity = 20.791972 – 21.628929 = -0.836957

The sensitivity of the call value to dividends occurs because dividends lower the stock price, reducing the potential upside. Because call options give the right to buy the stock at a fixed strike price, any drop in the stock price makes the call option less attractive, because it will be hard for the stock to trade greater than the strike price, that's why we have negative impact of dividends on call values.

On the other hand, put options give the right to sell the stock at a fixed strike price, and a dividend-induced drop in the stock price increases the probability that the stock will be below the strike price at expiration, making the put option more valuable. It increases the likelihood of the put being in-the-money, resulting in a higher payoff, thus have positive impact on put values.

**Problem2:**

Here we tried to implement using binomial tree method since we're dealing with American option with dividend payment, then we used a root scalar method to find the implied volatility for each portfolio. Then we can get the output like below:

| | Portfolio Value | Mean | VaR | ES | VaR (Delta-Normal) | \ |
|---|---|---|---|---|---|---|
| Call | 0.00 | -0.00 | 0.00 | 0.00 | 0.00 | |
| CallSpread | 0.00 | -0.00 | 0.00 | 0.00 | 0.00 | |
| CoveredCall | 165.00 | -0.19 | 11.53 | 14.30 | 12.08 | |
| ProtectedPut | 168.01 | -0.20 | 11.74 | 14.57 | 12.30 | |
| Put | 4.85 | -0.01 | 0.34 | 0.42 | 0.36 | |
| PutSpread | 3.01 | -0.00 | 0.21 | 0.26 | 0.22 | |
| Stock | 165.00 | -0.19 | 11.53 | 14.30 | 12.08 | |
| Straddle | 4.85 | -0.01 | 0.34 | 0.42 | 0.36 | |
| SynLong | -4.85 | 0.01 | -0.34 | -0.42 | -0.36 | |

| | ES (Delta-Normal) |
|---|---|
| Call | 0.00 |
| CallSpread | 0.00 |
| CoveredCall | 15.15 |
| ProtectedPut | 15.43 |
| Put | 0.45 |
| PutSpread | 0.28 |
| Stock | 15.15 |
| Straddle | 0.45 |
| SynLong | -0.45 |

---

Then we copied the codes from last week and apply the same inputs from this weeks, and get the result as following:

| | Mean | VaR | ES |
|---|---|---|---|
| Portfolio | | | |
| Call | 0.000000 | -0.000000 | -0.000000 |
| CallSpread | 0.000000 | -0.000000 | -0.000000 |
| CoveredCall | 0.261030 | 12.405701 | 15.488443 |
| ProtectedPut | -0.196251 | 10.142208 | 12.069244 |
| Put | -0.633287 | 3.259231 | 3.603648 |
| PutSpread | -0.303620 | 1.803272 | 2.043593 |
| Stock | 0.261030 | 12.405701 | 15.488443 |
| Straddle | -0.633287 | 3.259231 | 3.603648 |
| SynLong | 0.633287 | 3.079176 | 4.507139 |

As we can see from the numbers above, the normal distribution approach, VaR and ES, are higher for portfolios such as CoveredCall and Stock, meaning potential bigger losses. The reason is that this method assumes steady volatility and independent returns and doesn't account for changes in volatility over time which makes it more conservative.

On the other hand, the AR(1) model considers the way past returns can influence future returns, which often results in lower risk values. This method provides a more realistic measure of risks because it takes the previous price movements into consideration, thus will have more accurate numbers when asset prices are affected by previous price changes.

In Summary, the normal distribution approach is more conservative, while the AR(1) model is better for capturing real market behaviors to get a more accurate risk assessment.

**Problem3:**

For this problem, we're going to fit the data into a 4-factor model as shown in the class notes:

$$r_s - r_{rf} = \alpha + \beta_{mkt}\left(r_{mkt} - r_{rf}\right) + \beta_{SMB}SMB + \beta_{HML}HML + \epsilon_s$$

In 1997 Mark Carhart published an extension to Fama French adding a momentum factor. Momentum is the tendency of assets that have recently done well to continue doing well. The UMD (up minus down) factor is long high momentum stocks and short low momentum stocks.

$$r_s - r_{rf} = \alpha + \beta_{mkt}\left(r_{mkt} - r_{rf}\right) + \beta_{SMB}SMB + \beta_{HML}HML + \beta_{UMD}UMD + \epsilon_s$$

Then we found out the 20 interested stocks within the dataset and get the expected returns and calculate the covariance for them. After that, we tried to find the super efficient portfolio, which means that the portfolio will have the largest sharpe ratio.

We can get the super efficient portfolio as shown in the output:

Super Efficient Portfolio Weights:

| | Weight |
|---|---|
| AAPL | 0.00 |
| META | 7.87 |
| UNH | 14.23 |
| MA | 0.00 |
| MSFT | 0.00 |
| NVDA | 13.09 |
| HD | 0.00 |
| PFE | 0.00 |
| AMZN | 0.00 |

BRK-B 6.41

PG 16.75

XOM 0.00

TSLA 0.00

JPM 41.66

V 0.00

DIS 0.00

GOOGL 0.00

JNJ 0.00

BAC 0.00

CSCO 0.00

Sharpe Ratio: 3.2926575917867953